Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 96 (2016) 1249 – 1257

Procedia Computer Science

20th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2016, 5-7 September 2016, York, United Kingdom

Large Scale Microblogging Intentions Analysis with Pattern Based Approach

Mohamed Hamroun[a]*, Mohamed Salah Gouider[a], Lamjed Ben Said[a]

[a]*Institut Supérieur de Gestion, 41 Avenue de la liberté Bouchoucha, Le Bardo 2000, Tunisia*

Abstract

In recent years, social networks have become very popular. Twitter, a micro-blogging service, is estimated to have about 200 million registered users and these users create approximately 65 million tweets a day. Twitter constitutes a powerful medium today that people use to express their thoughts and intentions. The challenge is that each tweet is limited in 140 characters, and is hence very short. It may contain slang and misspelled words. Thus, it is difficult to apply traditional NLP techniques which are designed for working with formal languages, into Twitter domain. Another challenge is that the total volume of tweets is extremely high, and it takes a long time to process. In this paper, we describe a large-scale distributed system for intentions analysis process based on lexico semantic patterns using Hadoop Distributed File System (HDFS) and MapReduce functions. We conduct a case study of user intentions in the commercial field. The proposed method has stably performed data gathering and data loading. Besides, it has maintained stable load balancing of memory and CPU resources during data processing by the HDFS system. The proposed MapReduce functions have effectively performed intentions analysis in the experiments. Finally, obtained results show the importance and effectiveness of intentions detection using semantic patterns.

Peer-review under responsibility of KES International

Keywords: semantic patterns; intention analysis; microblogging; Hadoop; MapReduce; large scale

1. Introduction

Social networks services, such as Twitter and Facebook, constitute a powerful medium that allow users to publish over 400 million posts per day. The variety of social networks, and the use of digital and mobile is transforming the way business and customers interact. Many companies regularly use social networking websites to promote new products and services, and post announcements to the customers. On the other hand, users have a wide range to express their opinions and intentions towards products and services. The opportunity to capture user intention has raised growing interest both within the scientific community, leading to many exciting open challenges, as well as in the business world, due to the remarkable benefits to be had from marketing and financial prediction. Knowing user intentions can help merchants and advertisers promote their products and services online more accurately. Syntactic approaches including word and manual templates to extract user intentions have proven successful when applied to

E-mail address: mohamed.hamrounn@gmail.com

Peer-review under responsibility of KES International

doi:10.1016/j.procs.2016.08.169

product and politic reviews that contain well-structured sentences[1 2 3 4]. However, applying either approach to Twitter data faces several challenges. Due to the extensive use of abbreviations and irregular expressions in tweets, tweets data are often composed of poor grammatical sentences and syntactical structures[5]. Existing syntactic approaches to intention analysis mainly rely on parts of text. Intentions are explicitly expressed such as polarity terms, words, and their co-occurrence frequencies. In a previous work[6], we introduced a lexico semantic patterns of intentions analysis of Twitter. This work illustrated the creation of intentional ontology and its employment to enhance the patterns induction process. On a typical social media platform such as Twitter or Facebook, people consciously or unconsciously express their intentions all the time. If you issue the search query ”I want to buy” to the Twitter search engine, you will find a large number of Twitter posts (tweets) that express the desire or intention to buy all kinds of products. If you issue the query ”I want to watch,” you will find a large number of people who want to watch all kinds of movies and TV shows. Thus, it is necessary to develop technologies that rapidly extract meaningful information from the large amounts of data generated by social networks. A new real-time method is required to extract the intentions of users from this mass of data.

The contribution of this paper is twofold. The work presented here is an extension of previous work on intention analysis[6]. The research presented in does not deal with large volume of datasets due to the large volume of tweets. Thus, it takes a long time to process. Similarly, we identify intentions using lexico semantic patterns. In addition, we provide a new definition of fine-grained mining of intention components. We also discuss the way of porting patterns engine in a distributed architecture.

Recently, various open sources associated with the processing of big data have been provided. The Hadoop ecosystem[13] is a famous big data processing system that is most commonly used. More information on Hadoop will be offered in the next section. In this study, a parallel Hadoop Distributed File System (HDFS)[13] and MapReduce[14] functions are proposed. The proposed system can stably collect and store a variety of data generated by social networks to analyze user intentions.

2. Related Works

In this section, we briefly review previous studies on intention analysis

2.1. Previous works for intention analysis

In cognitive psychology an intention refers to the thoughts one has before producing an action. In regard to a system, a user intention is what the user expects a system to do. In many cases, we also talk about or write about our intentions. Intention analysis is a process to discover and extract user intention from the original data with the aid of text analytics, computational linguistics, and natural language processing. Up to now, a lot of researches have been developed to analyze the intentions of the users. Intention analysis has been considered as an information extraction problem[8 9]. Traditional supervised sequence learning methods such as conditional random fields (CRFs)[11] and hidden Markov models (HMMs) has been be applied. User intention has also been studied extensively in the commercial field.[10], intention classification is formulated as a two-class classification problem. Intention posts (positive class) are defined as posts that explicitly express a particular intention of interest. The other posts are treated as non-intention posts (negative class), although some of these posts may express some other kinds of intentions. In their experiments, the positive class was the intention to buy.[4] introduced a novel task of identifying wishes. A wish corpus composed by political comments and product reviews was constructed and studied in details. A mix of manual templates and SVM based text classifiers were applied on the wish corpus, and a method to identify more templates was also discussed.[3] interested in two specific kinds of wishes: suggestions about existing products and intentions that indicate the author will buy a product. The paper limited their research to product reviews. They thought a majority of the suggestion wishes had pivotal phrases involving modal verbs such as ”would”, ”could”, ”should” etc. So rules based on modal verbs were manually extracted.[1] studied the problem of automatically identifying wishes in product reviews. These wishes are sentences in which authors make suggestions about a product or show intentions to buy a product. This paper firstly proposed a keyword strategy to find candidate wish sentences. Then, sequential pattern are mined from these sentences that are manually labeled. Finally, by using patterns as features, a classifier is trained to identify wish sentences in product reviews.[6] introduced a novel method to automatically extract semantic

patterns for customer intentions analysis of Twitter. Customer Intention is designed by three key components (Holder, Intention Verb, Target). This work illustrated the creation of the domain ontology and its employment to enhance the patterns induction process. The basic idea behind the proposed approach is to take advantage of domain ontology for enhancing the pattern learning process regarding the knowledge contained in commercial tweets. When ontologies are employed in the patterns, potentially one pattern can describe multiple representations.

2.2. Intention Analysis on Big Data

Up to now, we are still unable to find studies related to intention analysis on Hadoop MapReduce architecture. However, there have been few works related to sentiment analysis on Hadoop.[16] described a large-scale distributed system for real-time sentiment analysis on Hadoop.[12] proposed a novel distributed algorithm implemented in Spark, an open source platform that translates the developed programs into MapReduce jobs. The proposed algorithm exploits the hash tags and emoticons inside a tweet, as sentiment labels, in order to avoid the time-intensive manual annotation task. In this study, a parallel HDFS that can stably extract and save the necessary data from a variety of social networks data is proposed. Moreover, we propose to transform algorithms both of patterns induction process as well as for the matching process in a parallel way using MapReduce.

3. Background

3.1. Intention definition

Our approach uses patterns to find intentions. Intentions detection is based on automatically scanning social methods posts for specific lexico semantic patterns. In our work, we propose a fine-grained mining of intention components where an intention is described as a triple (intended action, intention target, holder).

For example, in ”I plan to buy a camera,” the holder is I, the intended-action is to buy and the intention-target is a camera.

We make use of RDF ontology to store intentions. We formulated a pattern structure, which allows the use of classes from other semantic databases such as Wordnet[17], Verbnet[18] and DBPedia[19] in the construction of the patterns defining the intentions.

3.2. Hadoop MapReduce Implementation

MapReduce[14] architecture developed by Google was used with success on information retrieval tasks. Information extraction and pattern based annotation use similar methods such as information retrieval. This is another reason behind our decision to port our previous work into MapReduce architecture. Googles MapReduce[14] architecture seems to be a good choice for several reasons:

- Information processing tasks can benefit from parallel and distributed architecture with simply programming of Map and Reduce methods
- Architecture can process terabytes of data on PC clusters with handling failures
- Most information retrieval and information extraction tasks can be ported into MapReduce architecture, similar to pattern induction and matching algorithms.

The most accepted MapReduce implementation is Hadoop[13]. Hadoop controls the management of data on compute nodes by making use of the Hadoop Distributed File System (HDFS), scheduling the program’s execution over a set of machines, managing machine breakdowns, and handling the obligatory inter-machine communication.

4. MapReduce Functions for intentions analysis

4.1. Pattern Extraction

As described in a previous work, each pattern is described by a left hand side (LHS) and a right hand side (RHS). The LHS describes an intention representation and it consists of a subject (holder), relation (intentional verb) and an object (intention target). The subject and the object are the syntactic arguments of the relation, which describes the possible participations in the intention. In our implementation, the subject, object and the predicate are RDF classes that reside in the ontology. We denote the LHS of a pattern as follows:

$$(\$sub, \$I, \$obj) : RHS \tag{1}$$

The subject, relation, and object described in the LHS need to be identified in the RHS in order to provide a link between tweet and a new extracted fact. This can be done using labels, which are represented as words preceded by a "$" and followed by a colon and an equality sign, as well as a description of the attached token. Whenever the RHS matches with a sentence, the tokens with associated labels are filled in the LHS of the pattern.

$$(\$sub,\ kb:\ plan,\ \$obj) :\text{-}\ \$sub{:=}kb:\ I\ \$obj{:=}kb{:}camera \tag{2}$$

Note that "$I:" represents an intentional verb, which in our case refers to the ontology. The RHS on the right hand describes a pattern that has to be identified in tweets. We define a pattern as an ordered collection of tokens that are divided by spaces. Our approach supports a set of syntactic categories to describe the lexical category of the token. This step is defined a pre-processing stage. We distinguish between various verbs, nouns, adjectives, prepositions, coordinating conjunctions (e.g., "as well as") and cardinal numbers.

For distributing pattern extraction with MapReduce, each batch is processed independently by the mappers. No coordination is required between concurrent mappers. Thus the input to the mappers is tweets batches from the input corpus. The mapper scans the batch, one sentence at a time. If the mapper encounters a sentence with a pair of interesting entities, it emits triples of the form (e1, p, e2) along with the necessary part-of-speech information.

4.2. Pattern Matching

Multiple lexical representations describing the same intention may be derived from the same pattern. These representations are used in the pattern matching procedure. The pattern that is associated with a specific representation is retrieved. Then all the semantic classes are substituted by the participants that describe the third step substitutes both the participants (subject and object) and the relation for all the lexical representation by which they are denoted. The pattern matching is split into two phases, the Selection phase and the Join phase. Fig.1 and Fig.2 conceptually show the selection phase and the join phase respectively.

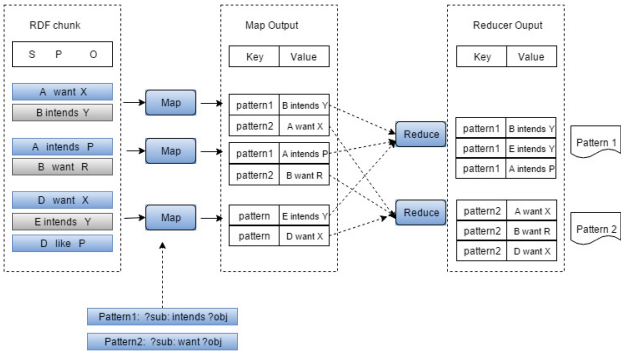

Fig. 1. Selection Phase.

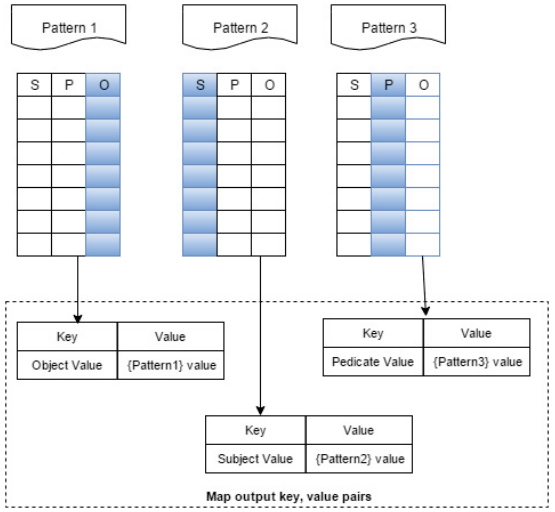

Fig. 2. Join Phase.

4.3. Pattern filtering

In this work, we wanted a fully automatic process for finding the best patterns that can identify intentions with high precision. Before being applied, patterns are evaluated and then filtered based on this evaluation. Patterns evaluation can be described in terms of a two-step process: First, a score score : $P \rightarrow \Re$ is assigned and then, potentially weakly performing patterns are filtered out by imposing a threshold or cut-off percentile on these scores. The scoring scheme for a pattern i is

$$pt_i = \frac{\sum_{k=1}^{m} \sqrt{freq(i,t_k)}}{\sum_{j=1}^{n} \sqrt{freq(i,t_j)}} \tag{3}$$

where m is the number of tweets with the DataSet that match the pattern, n is the number of all tweets that match the pattern, and freq (i, tk) is the number of times pattern i matched the tweet tk. We discard patterns that have weight less than a threshold (=0.5 in our experiments). After calculating scores for all the patterns, we choose the top K (=50 in our experiments) patterns.

Extracting patterns from large datasets is still time consuming when performing the computation on a single server. Thus we ported the pattern engine in a distributed architecture. In the next section, we discuss the porting of Patterns Engine into MapReduce architecture and its Hadoop implementation.

5. Patterns Engine porting to Hadoop MapReduce

In this section, the semantic patterns based approach for intention analysis is processed with the following steps using HDFS and MapReduce functions. First social networks data are gathered from social network services. Second, the necessary data is load into the HDFS. Third, the processed data is load into the parallel HDFS. Fourth, the intention analysis is processed via the MapRduce functions

5.1. Data Gathering

The data collection method of the proposed system was processed through Twitter. In addition to the acquisition of sample historical datasets, Appache Flume[20] has been used to retrieve data for continuous incremental data. Appache Flume is a data ingestion system that is configured by defining endpoints in a data flow called sources and sinks. Appache Flume decouples the source (Twitter) and the sink (HDFS) in this case. Both the source and the sink can

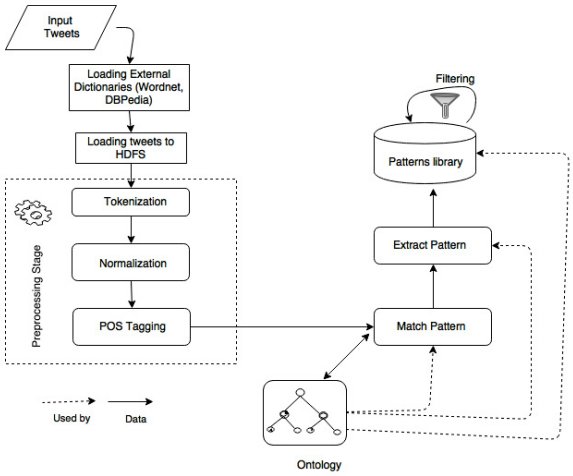

Fig. 3. The process of intention analysis.

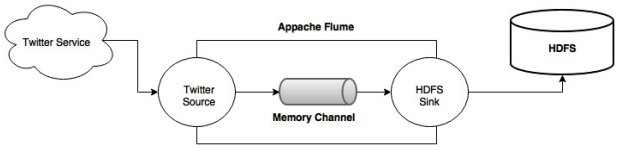

Fig. 4. Data gathering using Appache Flume.

operate at different speeds. It's also much easier to add new sources and sinks. Fig.4 shows the process of data gathering using Twitter API and Appache Flume.

5.2. Data Preprocessing

The data collected from Twitter contain a lot of unnecessary data. Thus, only the necessary data needs to be extracted from the collected data.

5.3. Intention analysis

The extracted data are stored in the proposed HDFS in parallel. Then, these data go through intention analysis via the MapReduce functions. Fig.3 illustrates the general framework of porting Patterns Engine on Hadoop Mapreduce.

6. RDF Generation

The generation of RDF out of the knowledge acquired by our approach is the final step of the extraction process. In previous work, semantic drift has been shown to be one of the key problems. In order to maintain a high precision and to avoid semantic drift within the proposed approach, we solely select the top-n percent of all scored patterns for generating RDF. We use information to calculate a confidence scores $s(t)$ for each triple t that we extract:

$$s(t) = \frac{1}{1 + e^{-[\sum_{i=1}^{n} s(p_i(t)]n+1}} \tag{4}$$

where $\sum_{i=1}^{n} s(p_i(t))$ is the sum of the score of all patterns that found the triple t and n the total number of patterns. Another challenge is that the growing size of our RDF database knowledge requires a new RDF representation to be scalable and high efficient. Therefore, based on[7] we used the distributed database model HBase.

7. Experimental Study

This section evaluates the effectiveness and the performance of the proposed system and discusses the results. The following three tests were carried out for performance analysis: performance test, time test, and accuracy test.

7.1. Experimental Enviroment

The experimental environment for performance analysis of the proposed system is described below. We use Hadoop 2.7.1. Our cluster consists of 5 machines running on Azure Microsoft using Ubuntu 12.04 as an operating system. Two mappers and two reducers are executed on each machine. The test for system load and acquisition time has been performed using the five Twitter data sets in Table 1. Each data set has been collected using Appache Flume and the Twitter API.

Table 1. Data sets for experiment and analysis.

DataSets	Number of tweets	Collection time(day)
Data1	2000	1
Data2	4500	2
Data3	30.000	9
Data4	70.000	20
Data5	120.000	55

7.2. Performence Test

Among the performance tests of the proposed system for data gathering and loading into HDFS, first was an experiment for system performance according to the number of data items. Fig.5 shows a comparison of HDFS loading time and crawling time for each data set. For data set "Data1," the collecting time was 40 seconds and the HDFS loading time was 5 second. For data set "Data5," the collecting time was 650 seconds and the HDFS loading time was 12 seconds, as shown in Fig.5. It is possible to see that the increases in HDFS loading time and collecting time are in proportion to the number of data items. Therefore, we can see that stable data collection and loading can be processed in a few seconds in the proposed system.

7.3. Performance Test for MapReduce Processing and Intention Analysis

A performance test for MapReduce processing and intention analysis was conducted. Intention analysis time and system load were tested according to the number of data items. The experiment was executed for the degree of system load and the time required for intentions analysis. Fig.6 shows the intentions analysis time required for each data set. Intention analysis took from 26 seconds to 68 seconds, in accordance with the scale of the data sets. Analysis time increases linearly with the scale of the data set, as shown in Fig.6. Our proposed system performed stably as the number of data items increased. The algorithm of the proposed method shows (n) processing time. It provides a stable parallel analysis environment without operating on a single node only.

7.4. Performance Test using Evaluation measures

In this section, we aim to evaluate the performance of our proposed approach in terms of precision (P), recall (R) and the F1 measure. These measures are suitable because our objective is to identify intention posts. Where Relevant is the set of relevant intention posts and Found is the set of found intention posts. There is a trade-off between precision and recall, and hence we compute the F1 measure. The F1 measure is applied to compute an even combination, i.e., the harmonic mean of precision and recall. These measurements are defined as follows:

$$P = \frac{|Relevant \bigcap Found|}{|Found|}, \qquad R = \frac{|Relevant \bigcap Found|}{|Relevant|}, \qquad R = \frac{2 \times P \times R}{P + R} \tag{5}$$

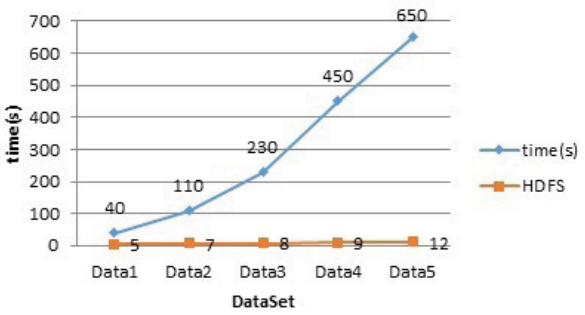

Fig. 5. Crawling time and HDFS loading time.

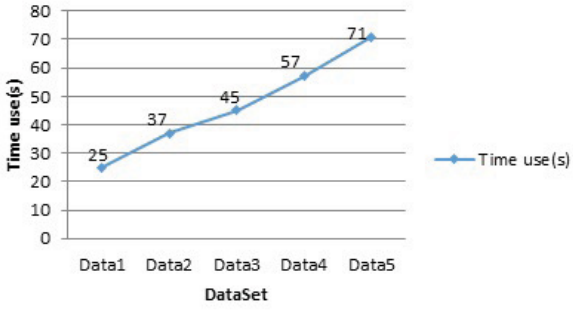

Fig. 6. Time spent for MapReduce processing and intention analysis.

Table 2. Precision, recall and F1 measures across 5 different datasets

DataSets	Data1	Data2	Data3	Data4	Data5
P	45%	58,81%	30%	57%	53,59%
R	55%	50%	58%	56,8%	55,59%
F	0.495	0.54	0.395	0.568	0.545

Table 2 shows the results across all datasets. The highest precision is achieved on the Dataset Data4 with 57%. While, the highest recall of 58% is obtained on the Data3 dataset.

8. Conclusions

This work proposed a new method to extract intention information from Twitter data using a parallel Hadoop Distributed File System (HDFS) to save data and using MapReduce functions for intentions analysis. Our approach does not rely on manual or limit syntactic templates of intentions detection. However it employs ontology concepts and relations designed by intentional verbs. We applied our approach on 5 different Twitter datasets. Experiments showed that our proposed system performed stably as the number of data items increased.

References

1. H. G'omez-Adorno, D. Pinto, M. Montes, G. Sidorov, and R. Alfaro, "Content and style features for automatic detection of users intentions in tweets," in Advances in Artificial IntelligenceIBERAMIA 2014. Springer, 2014, pp. 120128.
2. X. Wu and Z. He, "Identifying wish sentence in product reviews," Journal of Computational Information Systems, vol. 7, no. 5, pp. 1607 1613, 2011.
3. Ramanand, K. Bhavsar, and N. Pedanekar, "Wishful thinking: finding suggestions and'buy'wishes from product reviews," in Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text. Association for Computational Linguistics, 2010, pp. 5461.
4. A. B. Goldberg, N. Fillmore, D. Andrzejewski, Z. Xu, B. Gibson, and X. Zhu, "May all your wishes come true: A study of wishes and how to recognize them," in Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009, pp. 263271.
5. B. Hollerit, M. Kroll, and M. Strohmaier, Towards linking buyers and sellers: detecting commercial intent on twitter, in Proceedings of the 22nd international conference on World Wide Web companion. International World Wide Web Conferences Steering Committee, 2013, pp. 629632.
6. M. Hamroun, M. S. Gouider and L. B. Said, "Lexico Semantic Patterns for Customer Intentions Analysis of Microblogging," 2015 11th International Conference on Semantics, Knowledge and Grids (SKG), Beijing, 2015, pp. 222-226.
7. Sun, Jianling, and Qiang Jin. "Scalable rdf store based on hbase and mapreduce." Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on. Vol. 1. IEEE, 2010.
8. Sarawagi, Sunita. Information Extraction. Foundations and Trends in Databases, 2008. 1(3):261377.
9. Hobbs, Jerry R. and Ellen Riloff. Information Extraction. In Handbook of Natural Language Processing, 2nd ed., N. Indurkhya and F. J. Damerau, Editors. 2010: Chapman & Hall/CRC Press, 51132.

10. Chen, Zhiyuan, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Identifying Intention Posts in Discussion Forums. In Proceeding of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2013). 2013c.
11. Lafferty, John, Andrew McCallum, and Fernando Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of International Conference on Machine Learning (ICML-2001). 2001.
12. Nodarakis, Nikolaos, et al. "Large Scale Sentiment Analysis on Twitter with Spark."
13. Hadoop, http://hadoop.apache.org/.
14. Dean, J.Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters.Google, Inc., OSDI 04, San Francisco, CA, December 2004.
15. L. George, HBase:The Definitive Guide, OReilly, 2011.
16. V. N. Khuc, C. Shivade, R. Ramnath, and J. Ramanathan, Towards building large-scale distributed systems for Twitter sentiment analysis, in Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC 12), pp. 459464, March 2012.[24] M. Bautin, C. B.
17. Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database*. International journal of lexicography, 3(4), 235-244.
18. Schuler, K. K. (2005). VerbNet: A broad-coverage, comprehensive verb lexicon. ISO 690
19. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007). Dbpedia: A nucleus for a web of open data (pp. 722-735). Springer Berlin Heidelberg.
20. Hoffman, S. (2013). Apache Flume: Distributed Log Collection for Hadoop. Packt Publishing Ltd.